Docket No.: 50325-0841

AMENDMENTS TO THE CLAIMS

Please amend Claims 1, 2, 3, 5, 6, 7, 9, 12-17, 19, 20, 21, 23, and 26 as follows.

1. (Currently Amended) A method of a development and build environment for packaged software delivery in a distributed network of nodes, the method comprising the computer-implemented steps of:

compiling source code files into executable file modules;

wherein a module contains an image for a process or a dynamically linked library (DLL);

creating a software package that comprises at least one module;

wherein packages are created based on features/characteristics or purpose;

creating metadata for a module that includes~~, but is not limited to,~~ any module information such as the module's: binary signature, name, directory path, and characteristics;

inserting the module's metadata into the software package;

gathering application program interface (API) dependency information for each module;

wherein a module can provide and use at least one API;

collecting dependencies documented in module specifications and placing them into the module's metadata; and

wherein the dependencies documented in each module lists API names and versions that the process or DLL depends on.

2. (Currently Amended) A method as recited in Claim 1, ~~further comprising the step of: providing a linker; and~~

wherein ~~said~~ a linker creates a list of dependent modules for a given process or DLL module and places the list in the module's metadata.

3. (Currently Amended) A method as recited in Claim 1, further comprising the steps of:

creating metadata for each API;

inserting the API metadata into the software package; and

wherein metadata for an API includes, but is not limited to: the API's name and version.

4. (Original) A method as recited in Claim 1, further comprising the step of:

calculating a binary signature for each module and inserting the binary signature into the respective module's metadata; and

wherein each unique version of a module will have a unique binary signature.

5. (Currently Amended) A method as recited in Claim 1, further comprising the steps of:

creating metadata for a package that includes~~, but is not limited to,~~ any package information such as the package's: name, build date, and characteristics; and

inserting the package metadata into the package.

6. (Currently Amended) A method of a development and build environment for packaged software delivery in a distributed network of nodes, the method comprising the computer-implemented steps of:

compiling source code files into executable file modules;

wherein a module contains an image for a process or a dynamically linked library (DLL);

creating a software package that comprises at least one module;

creating metadata for a module that includes~~, but is not limited to,~~ any module information such as the module's: binary signature, name, directory path, and characteristics;

inserting the module's metadata into the software package;

gathering application program interface (API) dependency information for each module; and

wherein a module can provide and use at least one API.

7. (Currently Amended) A method as recited in Claim 6, ~~further comprising the step of providing a linker; and~~

wherein ~~said~~ a linker creates a list of dependent modules for a given process or DLL module and places the list in the module's metadata.

8. (Original) A method as recited in Claim 6, further comprising the step of:

collecting dependencies documented in module specifications and placing them into the module's metadata; and

wherein the dependencies documented in each module lists API names and versions that the process or DLL depends on.

9. (Currently Amended) A method as recited in Claim 6, further comprising the steps of:

creating metadata for each API;

inserting the API metadata into the software package; and

wherein metadata for an API includes, but is not limited to: the API's name and version.

10. (Original) A method as recited in Claim 6, further comprising the step of:

calculating a binary signature for each module and inserting the binary signature into the respective module's metadata; and

wherein each unique version of a module will have a unique binary signature.

11. (Original) A method as recited in Claim 6, wherein packages are created based on features/characteristics or purpose.

12. (Currently Amended) A method as recited in Claim 6, further comprising the steps of:

creating metadata for a package that includes~~, but is not limited to,~~ any package information such as the package's: name, build date, and characteristics; and

inserting the package metadata into the package.

13. (Currently Amended) An apparatus for a development and build environment for packaged software delivery in a distributed network of nodes, comprising:

means for compiling source code files into executable file modules;

wherein a module contains an image for a process or a dynamically linked library (DLL);

means for creating a software package that comprises at least one module;

means for creating metadata for a module that includes~~, but is not limited to,~~ any module information such as the module's: binary signature, name, directory path, and characteristics;

means for inserting the module's metadata into the software package;

means for gathering application program interface (API) dependency information for each module; and

wherein a module can provide and use at least one API.

14. (Currently Amended) An apparatus as recited in Claim 13, further comprising ~~the step of~~:

a linker; and

wherein said linker creates a list of dependent modules for a given process or DLL module and places the list in the module's metadata.

15. (Currently Amended) An apparatus as recited in Claim 13, further comprising ~~the step of~~:

means for collecting dependencies documented in module specifications and placing them into the module's metadata; and

wherein the dependencies documented in each module lists API names and versions that the process or DLL depends on.

16. (Currently Amended) An apparatus as recited in Claim 13, further comprising ~~the step of~~:

means for creating metadata for each API;

means for inserting the API metadata into the software package; and

wherein metadata for an API includes, but is not limited to: the API's name and version.

17. (Currently Amended) An apparatus as recited in Claim 13, further comprising ~~the step of~~:

means for calculating a binary signature for each module and inserting the binary signature into the respective module's metadata; and

wherein each unique version of a module will have a unique binary signature.

18. (Original) An apparatus as recited in Claim 13, wherein packages are created based on features/characteristics or purpose.

19. (Currently Amended) An apparatus as recited in Claim 13, further comprising ~~the step of~~:

means for creating metadata for a package that includes~~, but is not limited to,~~ any package information such as the package's: name, build date, and characteristics; and

means for inserting the package metadata into the package.

20. (Currently Amended) A computer-readable medium carrying one or more sequences of instructions for a development and build environment for packaged software delivery in a distributed network of nodes, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of:

compiling source code files into executable file modules;

wherein a module contains an image for a process or a dynamically linked library (DLL);

creating a software package that comprises at least one module;

creating metadata for a module that includes~~, but is not limited to,~~ any module information such as the module's: binary signature, name, directory path, and characteristics;

inserting the module's metadata into the software package;

gathering application program interface (API) dependency information for each module; and

wherein a module can provide and use at least one API.

21. (Currently Amended) A computer-readable medium as recited in Claim 20, ~~further comprising the step of:~~

~~providing a linker; and~~

wherein ~~said~~ a linker creates a list of dependent modules for a given process or DLL module and places the list in the module's metadata.

22. (Original) A computer-readable medium as recited in Claim 20, further comprising the step of:

collecting dependencies documented in module specifications and placing them into the module's metadata; and

wherein the dependencies documented in each module lists API names and versions that the process or DLL depends on.

23. (Currently Amended) A computer-readable medium as recited in Claim 20, further comprising the steps of:

creating metadata for each API;

inserting the API metadata into the software package; and

wherein metadata for an API includes, but is not limited to: the API's name and version.

24. (Original) A computer-readable medium as recited in Claim 20, further comprising the step of:

calculating a binary signature for each module and inserting the binary signature into the respective module's metadata; and

wherein each unique version of a module will have a unique binary signature.

25. (Original) A computer-readable medium as recited in Claim 20, wherein packages are created based on features/characteristics or purpose.

26. (Currently Amended) A computer-readable medium as recited in Claim 20, further comprising the steps of:

creating metadata for a package that includes~~, but is not limited to,~~ any package information such as the package's: name, build date, and characteristics; and

inserting the package metadata into the package.